

# V3 HCR-FISH Image Analyses

Melissa Miller

## Contents

<b>Overview</b>	<b>2</b>
<b>Image acquisition</b>	<b>2</b>
<b>Image processing</b>	<b>2</b>
Tissue isolation . . . . .	2
Overview . . . . .	2
Steps . . . . .	2
Segmentation . . . . .	13
Overview . . . . .	13
Image pre-processing . . . . .	13
Cellpose to get cell and/or nuclear outlines . . . . .	13
Locally . . . . .	13
On a remote server . . . . .	14
<b>Image analyses</b>	<b>15</b>
Dot detection with RS-FISH . . . . .	15
Overview . . . . .	15
Image pre-processing . . . . .	15
RS-FISH FIJI plugin to identify dots . . . . .	15
Using the ROI converter to measure single-cell level expression . . . . .	26
Cellpose single-image ROI converter . . . . .	26
Running the batch ROI converter . . . . .	43
What to do now . . . . .	44
Statistics . . . . .	44
Expression intensity along the proximodistal axis . . . . .	44
Overview . . . . .	44
Steps . . . . .	44
Colocalization . . . . .	49
Statistics . . . . .	49

## Overview

This notebook walks through the steps to measure gene expression using V3 HCR-FISH.

### Image acquisition

### Image processing

#### Tissue isolation

##### Overview

Below you will find the steps that need to be run for each of your raw images to spatially isolate tissues into different areas of interest. It will make your life a lot easier down the line if files are named consistently, including capitalization, and without spaces or special characters.

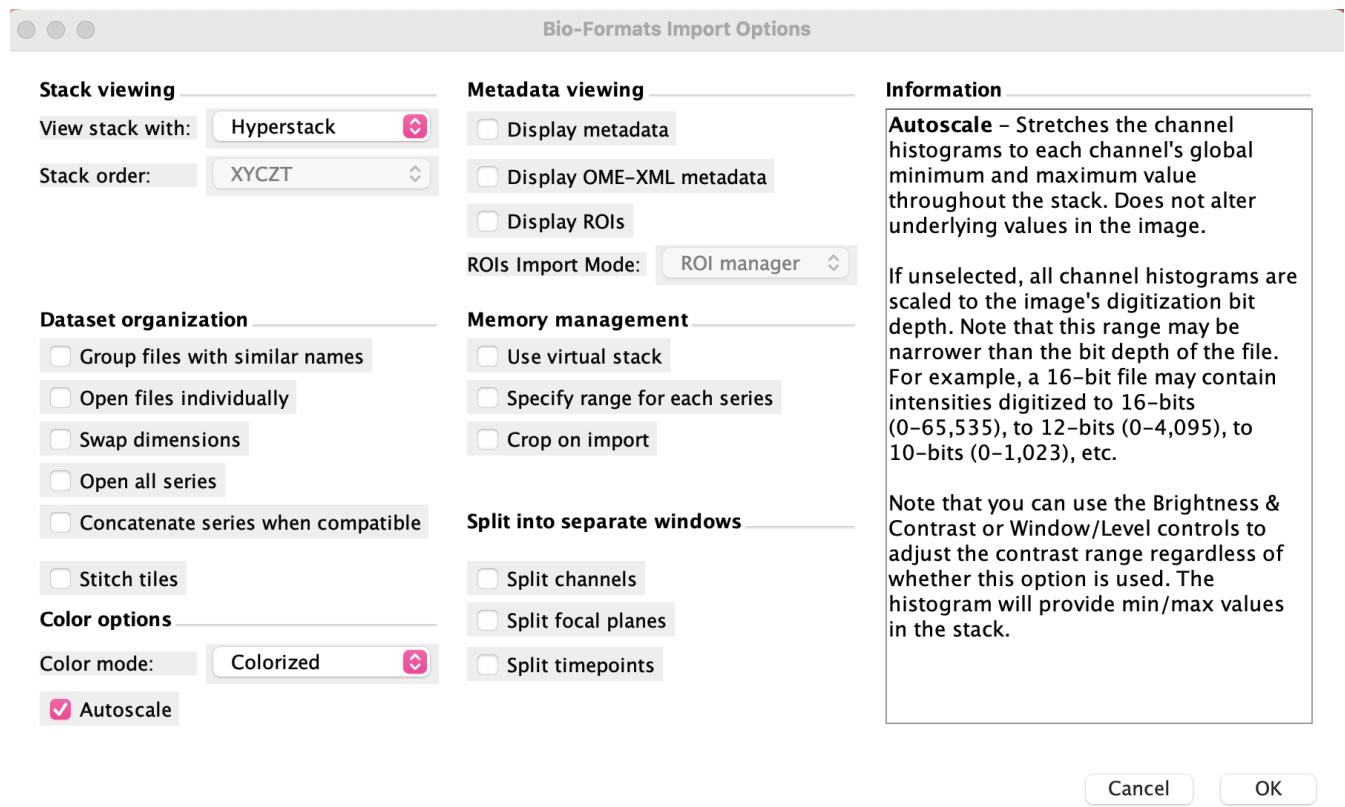
**DO NOT** crop your images or otherwise change their dimensions when working with multiple tissues from an original image (zooming in and out is fine). This will impact how the coordinates of the segmentation results line up with the cells and nuclei in the image.

##### Steps

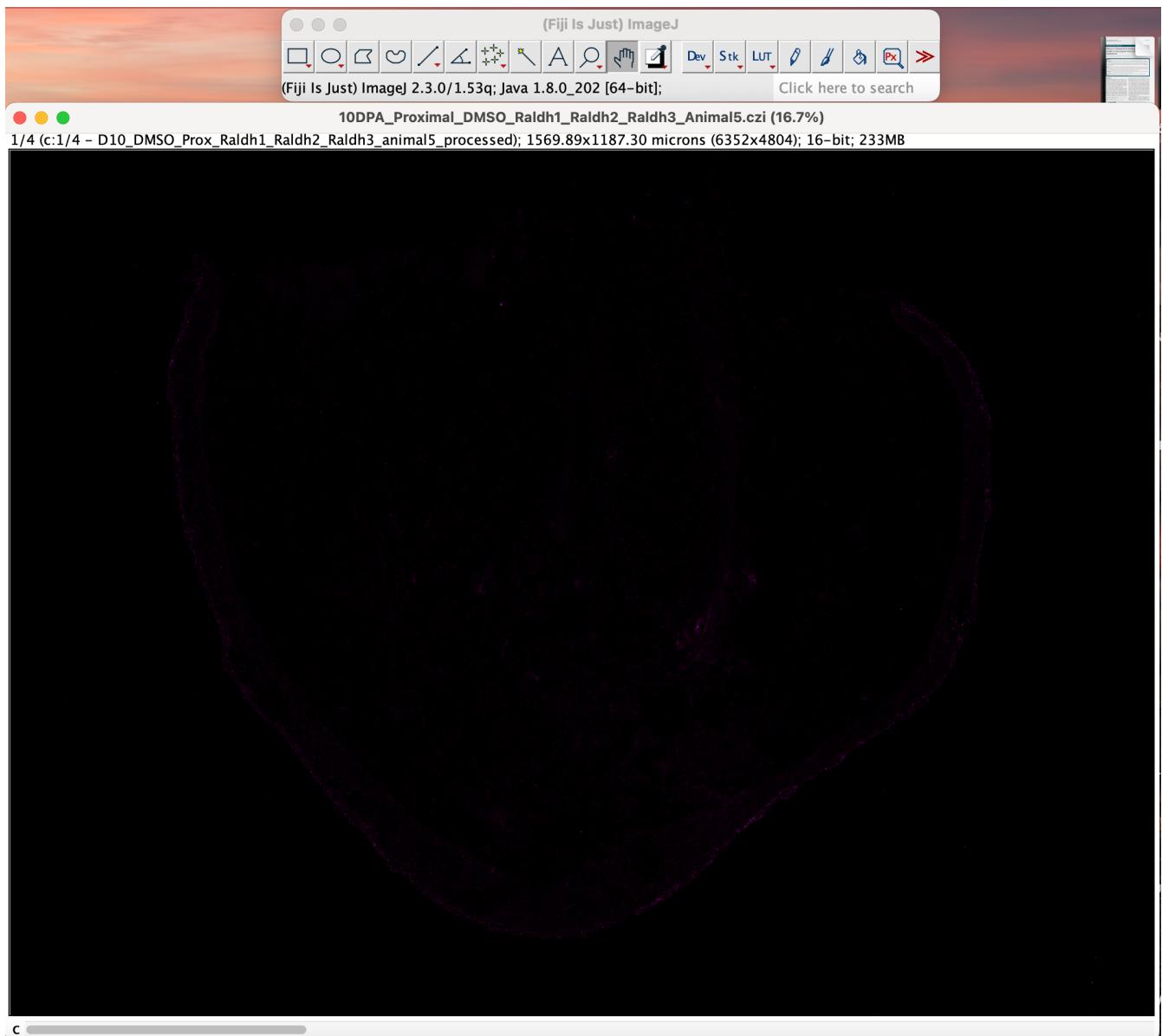
In this example, I'll be starting with `10DPA_Proximal-DMSO_Raldh1-Raldh2-Raldh3_Animal5.czi`. After a three-gene v3 HCR-FISH with a DAPI nuclear stain, this image was taken on a Zeiss LSM 880 with 4 channels, 5 z-planes, and tiling. Prior to this current step, the image was Airyscan processed on the Zen Black software, then a maximum intensity projection was performed and the tiles stitched to produce a single-plane, four-channel image.

##### Inspecting your image

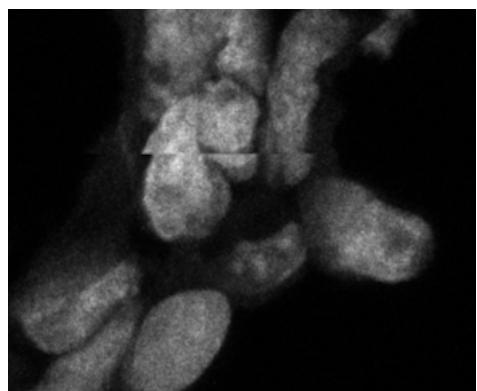
1. Open the image with FIJI. If a screen pops up with `Bio-Formats Import Options` at the top, change the settings to match these:



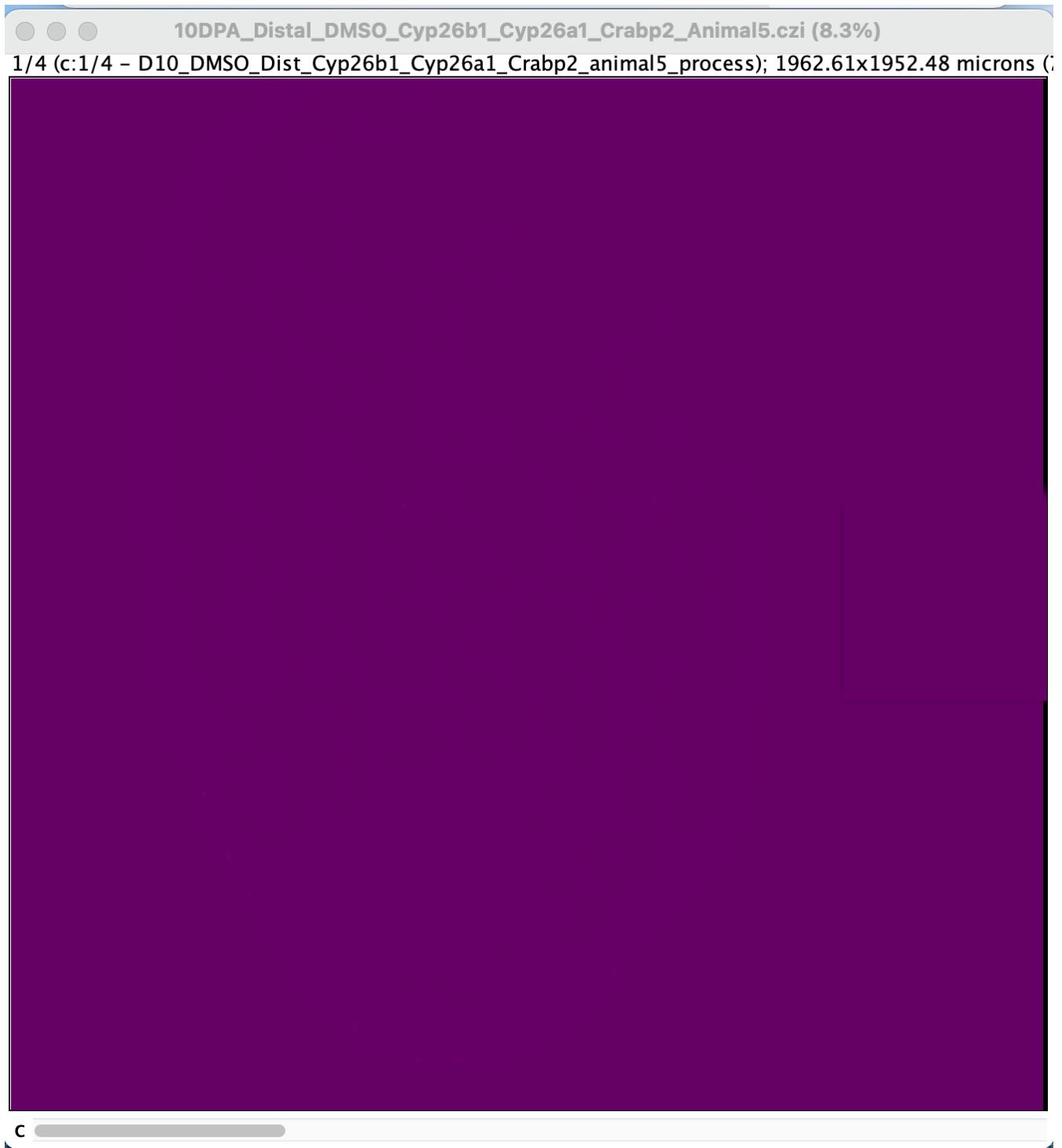
2. Click OK. You can ignore warning messages that pop up in the Console window. The colors might not be very bright, and that's okay. You can adjust the brightness for visualization with the **Image > Adjust > Brightness & Contrast** window in FIJI. Don't click **Apply** - this will change the actual pixel intensities.



3. Using the scroll bar below the image, look for any issues with stitching. This should be pretty apparent in the cytoplasm or nuclear stain channel, and will appear as a seam on the image like this:

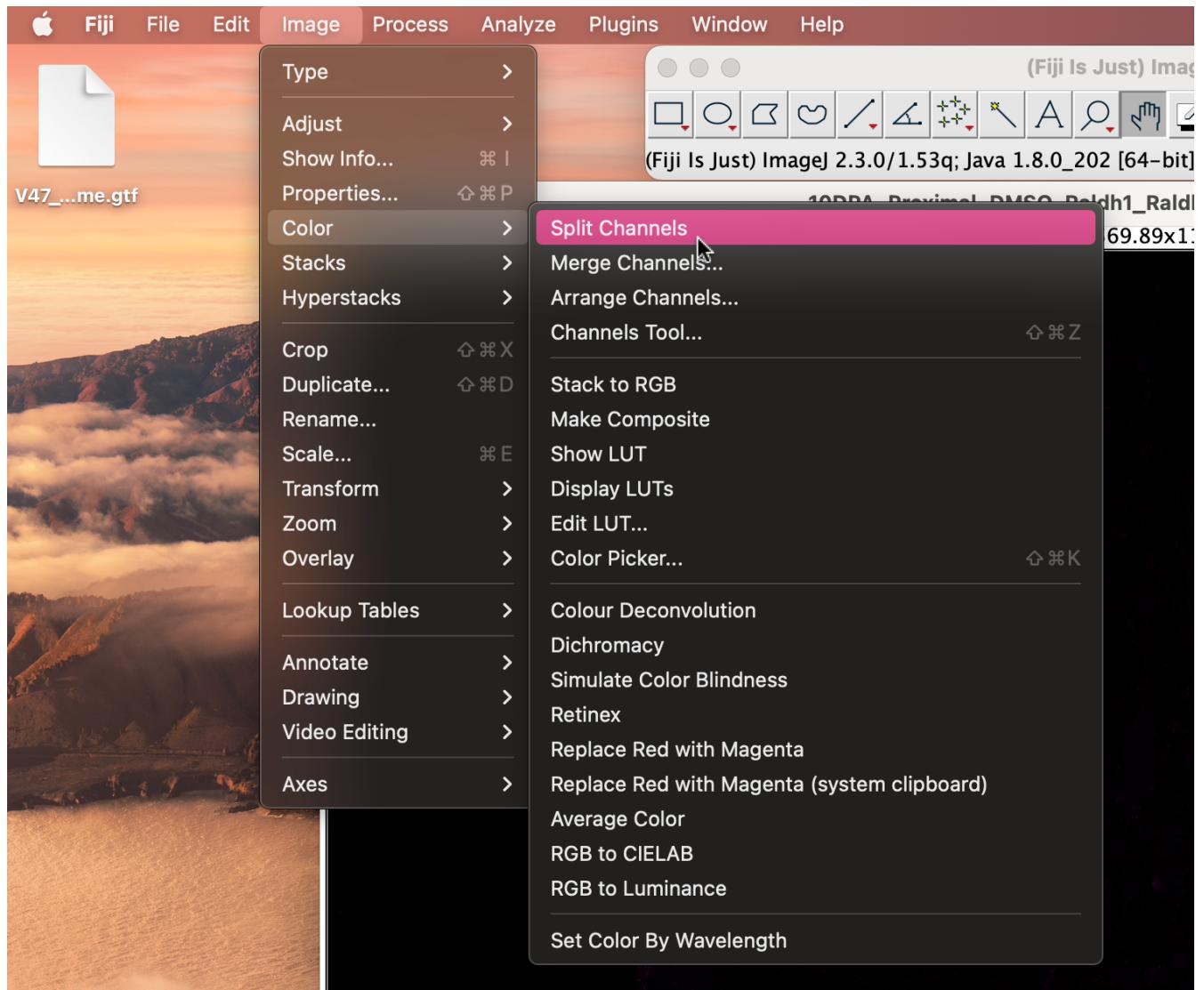


If this stitching error is not in an area of the tissue that you are interested in measuring, and the image has no other issues, it can be ignored; otherwise, do not go further with this image - it will need to be re-stitched. If you encounter the following view with or without evident seams like below, this is **not a problem** again if the bad stitch is not on the tissue (note the faint tile boundaries in the right-center tile):

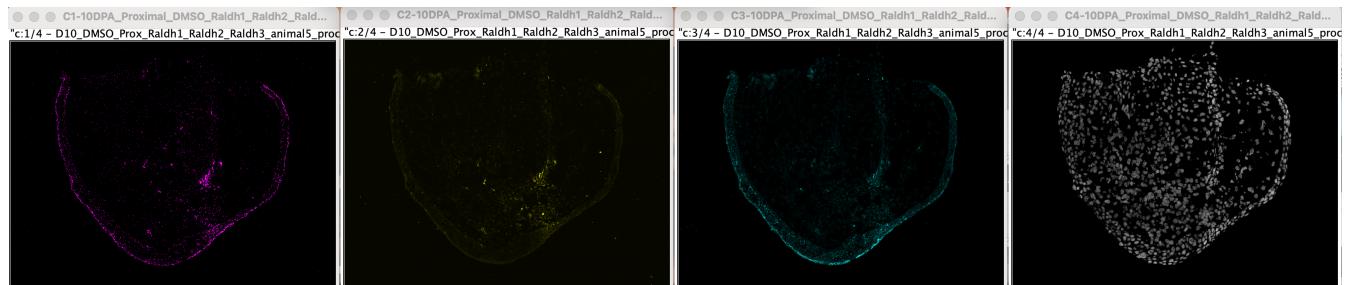


Split the image channels in multi-channel images

4. Split the image channels:



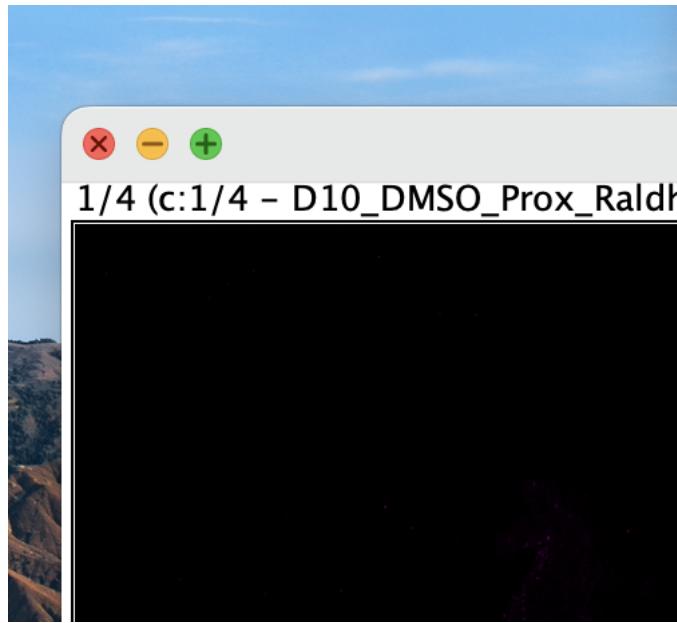
You should end up with several new image windows, one for each channel from your image. Here are some side-by-side (edited to show up a little brighter):



5. Save each window from step 4 as a new .tif (File > Save As > Tiff...). Name each of these images with the original file name, e.g. 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5, but

add \_Whole-Tissue\_GeneName.tif at the end. FIJI will automatically add a channel number like C1- to the beginning of each file, so be sure to also delete this. GeneName should align with whatever gene was imaged in that channel, chosen before starting HCR-FISH.

If you forget which color aligns with which channel, look in the top left corner of the image. There should be an indication of 1/4, 2/4, 3/4, or 4/4:



These correspond to the channels, which for our images happen to be the gene names in order. In this example, with 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5.czi, this means that 1/4 is Raldh1, 2/4 is Raldh2, 3/4 is Raldh3, and 4/4 is DAPI.

**At the end of step 5** you should now have one new .tif file for each channel of your original image. In this example, they are:

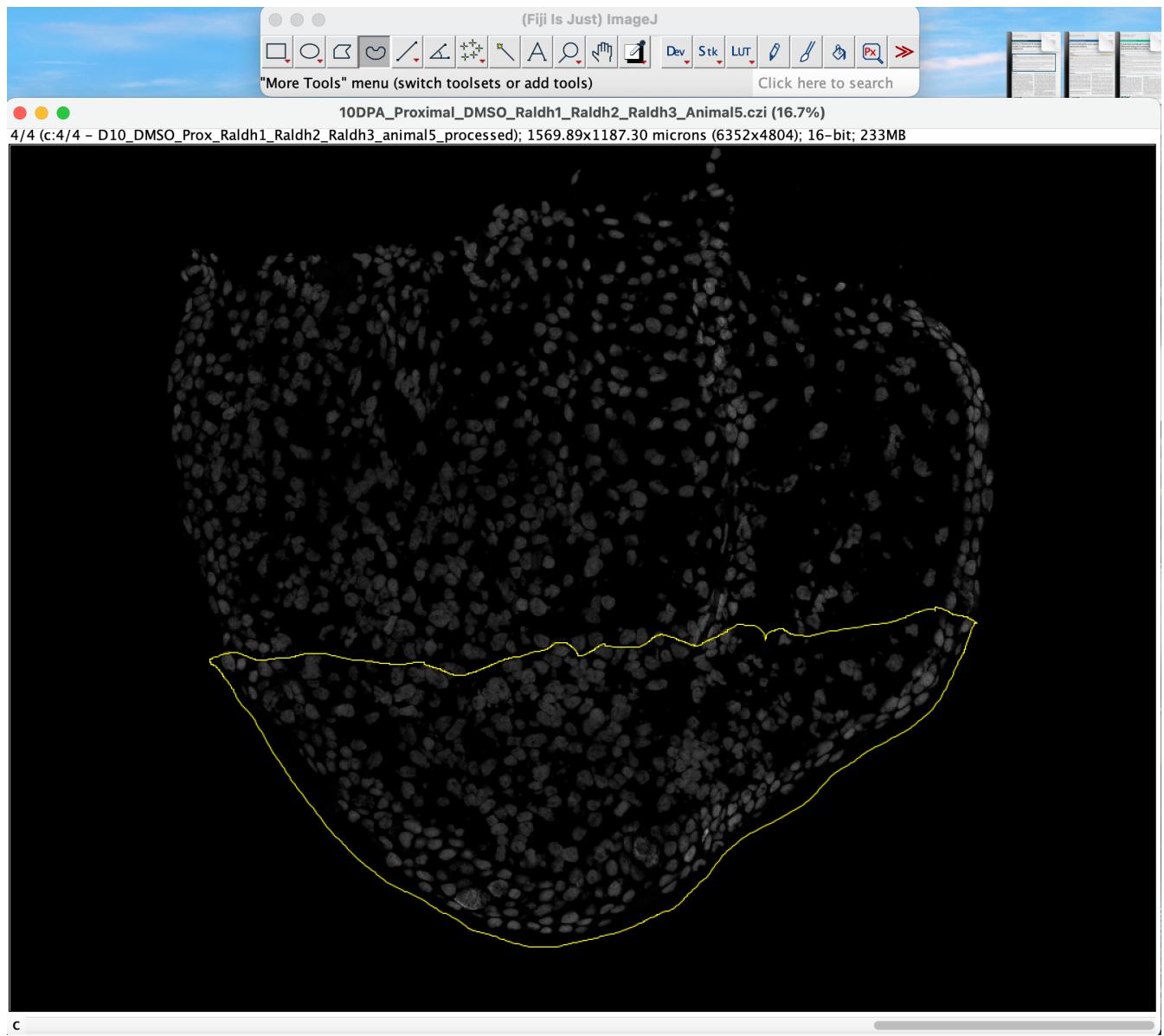
- 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5\_Whole-Tissue\_Raldh1.tif
- 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5\_Whole-Tissue\_Raldh2.tif
- 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5\_Whole-Tissue\_Raldh3.tif
- 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5\_Whole-Tissue\_DAPI.tif

Close all of these except for the image corresponding to your nuclear or cytoplasm stain.

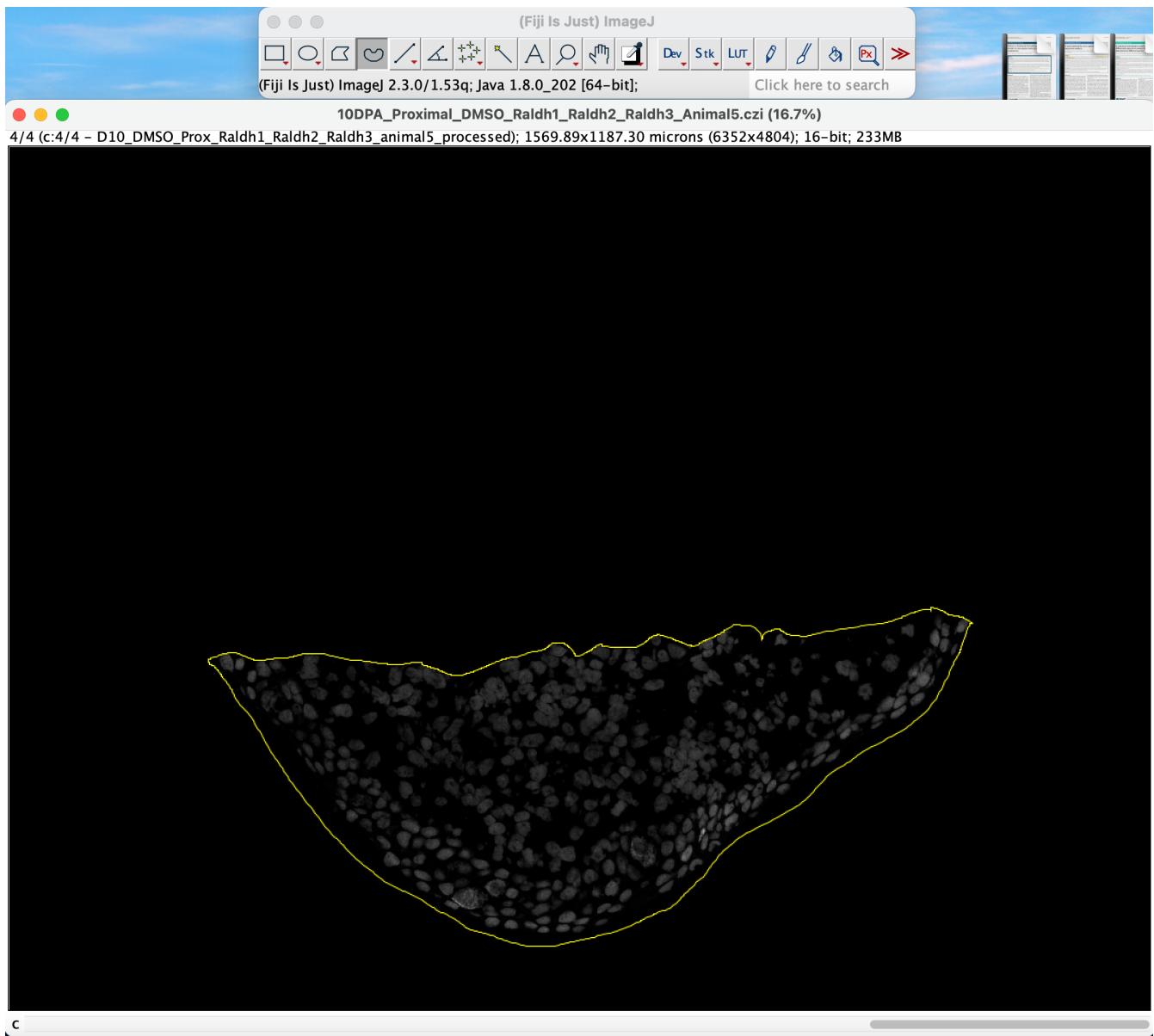
#### Isolate areas of interest using the freehand tool

6. Using the freehand selection tool in FIJI, draw around one area of the tissue you are interested in, trying your best to avoid cutting through cells / nuclei. In the example below, I am isolating blastema tissue at the amputation plane, as the rest of the limb tissue is not pertinent to our work.





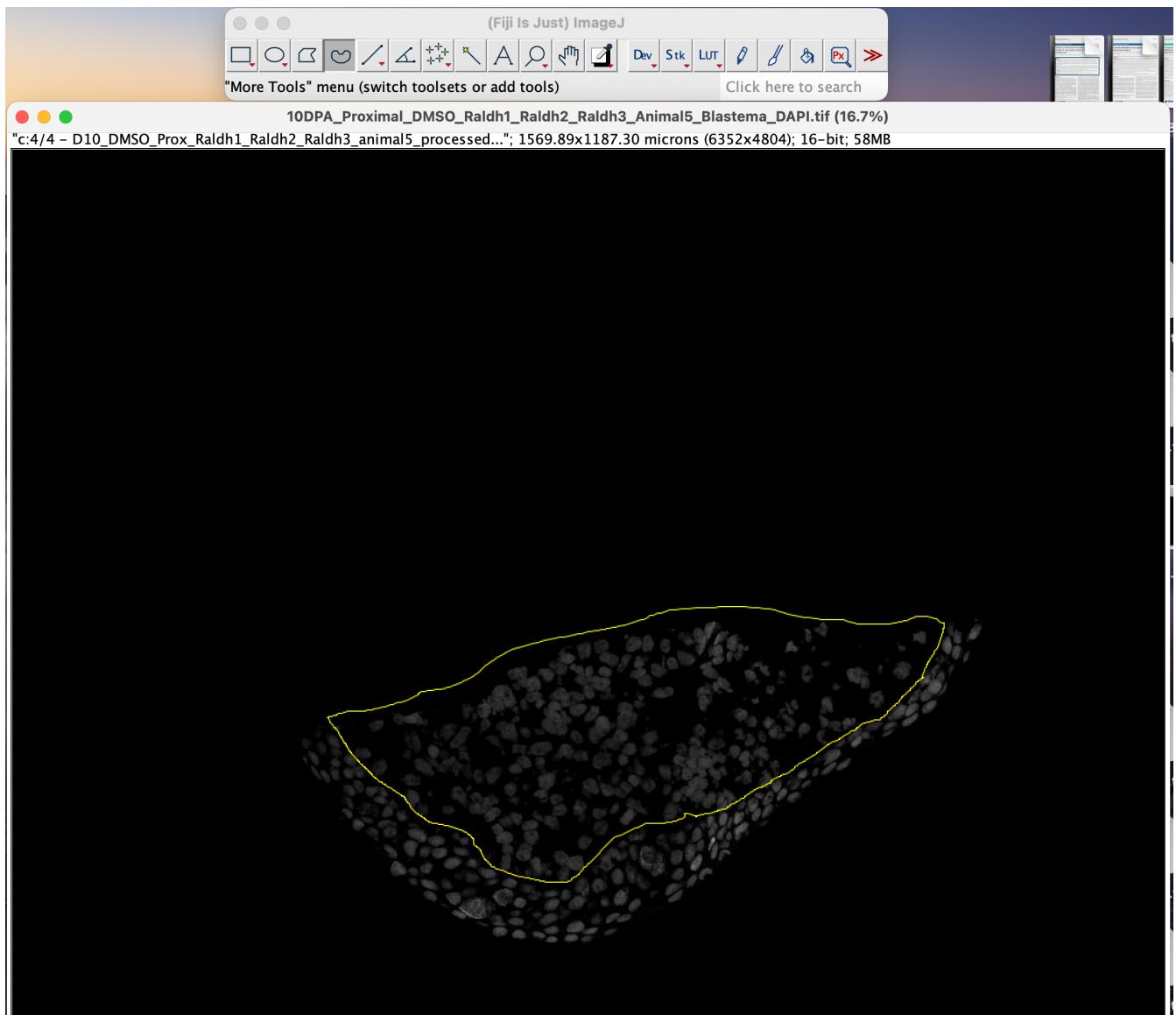
7. Click **Edit > Clear Outside** to remove anything outside of the boundary.



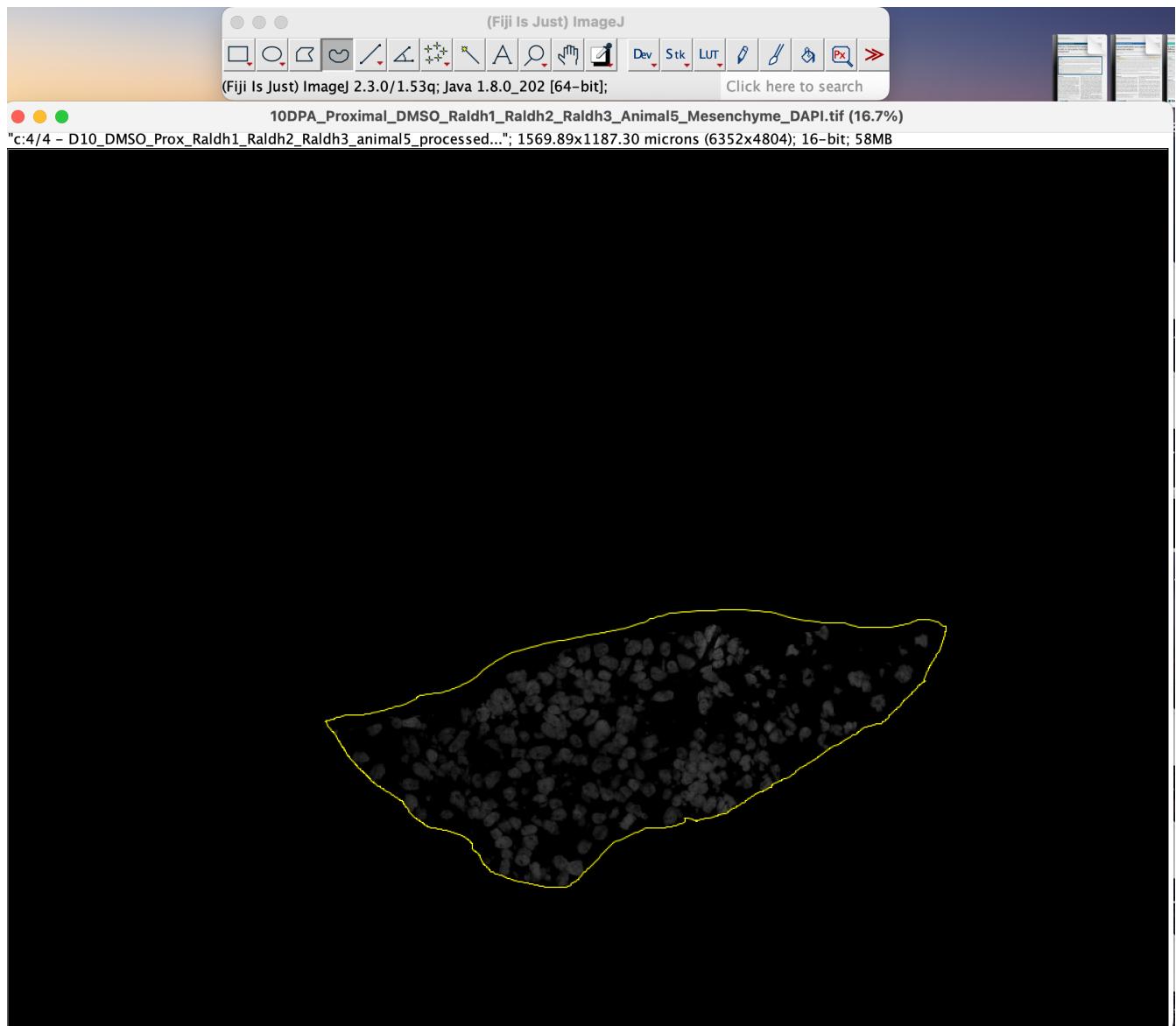
8. Save this image as a new .tif with the original file name, but add \_{Tissue}\_{Stain}.tif to the end, where Tissue is whatever area the selection was and Stain being the cytoplasm or nuclear stain used. In this example, this is 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5\_Blastema\_DAPI.tif.
- If you would like to isolate further areas, you can go to Edit > Undo and the tissue outside of the boundary will reappear; the image you saved above will still have the tissue isolated. You can redo this step as many times as desired, drawing a selection around different tissues of interest and clearing the remaining tissue, saving the specific tissue files as you go along. If your tissue of interest is, for example, everything **but** what you just isolated, and your selection boundary is gone, you can go to Edit > Selection > Restore Selection, and then Edit > Clear. An example of this is below with two other tissues we are interested in.

#### Example of more tissues

9. Using the freehand tool, draw along the inner edge of the epithelium and around the mesenchyme:

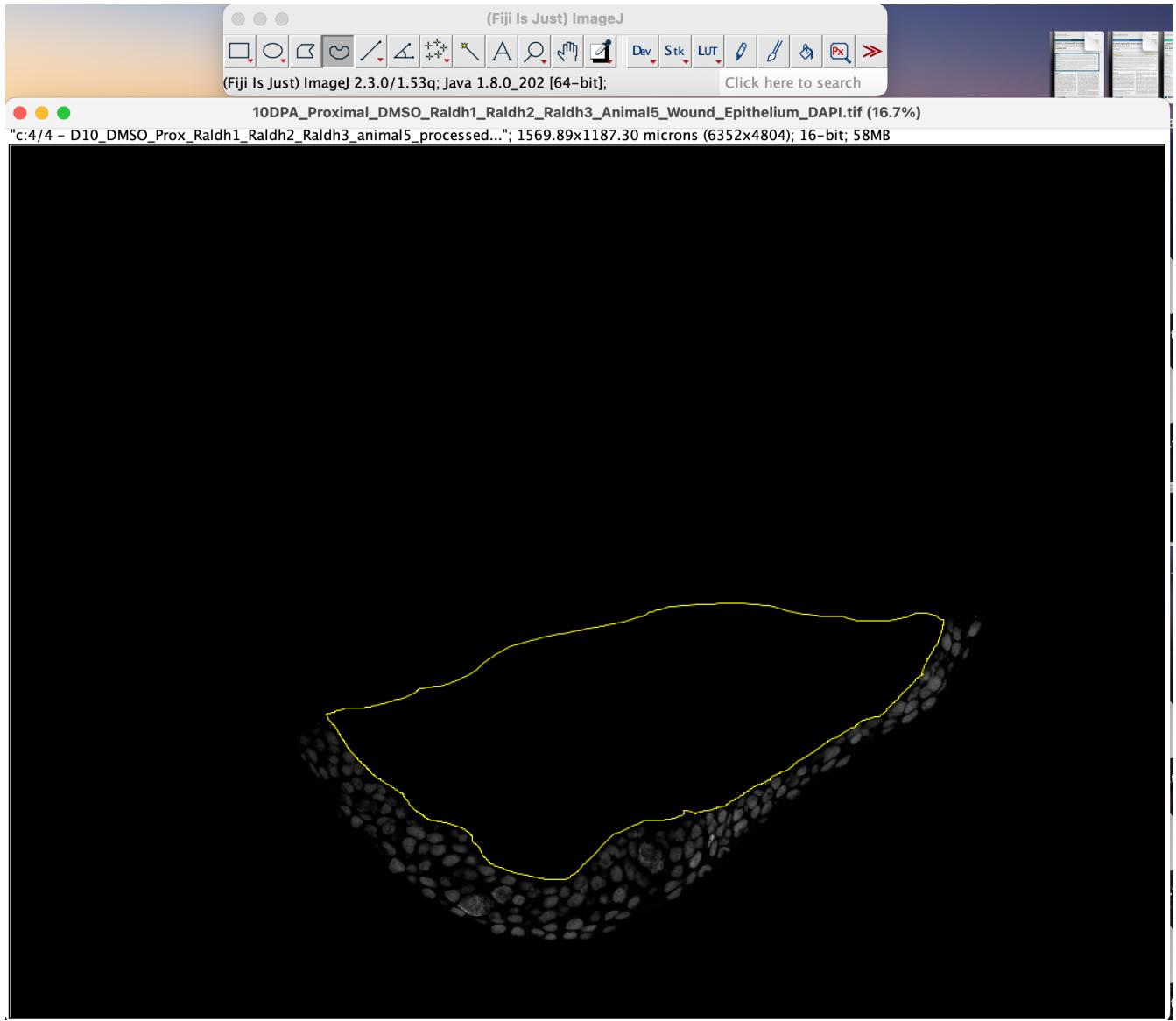


10. Use `Clear Outside`, then save this image as a new `.tif` with the original file name, adding `_Mesenchyme_DAPI.tif` to the end.



In this example, this is `10DPA_Proximal-DMSO_Raldh1-Raldh2-Raldh3_Animal5_Mesenchyme_DAPI.tif`.

11. Use **Edit > Undo**, then use **Clear**. Save this as a new .tif, with `_Epithelium_DAPI.tif`.



In this example, this is `10DPA_Proximal-DMSO_Raldh1-Raldh2-Raldh3_Animal5_Epithelium_DAPI.tif`.

**Resulting output** After all of these steps, you should now have the following images:

1. The **original** unaltered image in `.czi` format. There will be nothing further done with this image, so you can go ahead and store it somewhere.
  - Ex: `10DPA_Proximal-DMSO_Raldh1-Raldh2-Raldh3_Animal5.czi`.
2. New **Whole\_Tissue** images, one per channel from the original image, in `.tif` format.
  - `10DPA_Proximal-DMSO_Raldh1-Raldh2-Raldh3_Animal5_Whole-Tissue_DAPI.tif`
  - `10DPA_Proximal-DMSO_Raldh1-Raldh2-Raldh3_Animal5_Whole-Tissue_Raldh1.tif`
  - `10DPA_Proximal-DMSO_Raldh1-Raldh2-Raldh3_Animal5_Whole-Tissue_Raldh2.tif`
  - `10DPA_Proximal-DMSO_Raldh1-Raldh2-Raldh3_Animal5_Whole-Tissue_Raldh3.tif`

3. New images of each isolated tissue you are interested in, in .tif format.

- 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5\_Blastema\_DAPI.tif
- 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5\_Mesenchyme\_DAPI.tif
- 10DPA\_Proximal-DMSO\_Raldh1-Raldh2-Raldh3\_Animal5\_Epithelium\_DAPI.tif ## Segmentation

## Segmentation

### Overview

This pipeline is designed to provide single-cell resolution gene expression data to the qualitative, visual data generated by v3 HCR-FISH. This pipeline assumes basic familiarity with the command line, Anaconda package manager, and Jupyter Notebook or a text editor like Vim.

### Image pre-processing

After imaging you should have already processed the raw images with AiryScan processing, stitching, and max intensity projection. Then you will need to divide your processed whole-tissue, multi-channel image to separate out tissue of interest (blastema, epithelium, and mesenchyme, for example) as well as split the color channels. The instructions for doing this are here.

### Cellpose to get cell and/or nuclear outlines

Cellpose is a machine-learning algorithm designed for cell and nucleus segmentation in both 2D and 3D. There is support for a GUI as well as use in the command line and Jupyter Notebook. It also has documentation and a detailed GitHub page that makes it easy to navigate.

### Locally

**Installation on Windows** Follow the instructions here for how to install Cellpose on a Windows or Linux system, including if you want GUI support or not.

**Note: Mac (as of October 2022)** There is a dependency of Cellpose that is currently too out-of-date to be compatible with the newest Mac software. If you follow the instructions above for a Mac/Linux system, you will successfully install Cellpose, but it will not be able to utilize your computer's GPU, and instead will resort to CPUs (which it will tell you if you try to run it in the command line or a Jupyter Notebook). Cellpose will attempt to run, but it could take upwards of an hour to segment one image, and eat up enough RAM in the process that it could crash your machine. If you have a Mac, and this dependency issue is not yet resolved, I recommend to instead use a computing cluster.

**The GUI** If you installed Cellpose with the GUI above, you will have the ability to open a desktop application to run Cellpose through. This is good for when you only have a few images, but as your numbers grow, you'll want to consider automating the segmentation in something like Jupyter or in a script file. Visit the Cellpose documentation for more information about the GUI.

The most important thing is that, when Cellpose is done segmenting and you're happy with the results, go to **File** in the top-left and **save the text outlines** of the segmentation. No other output is necessary for this pipeline.

**Jupyter Notebook** There is an example notebook for running Cellpose segmentation here. I also have a Notebook for running Cellpose in a loop here. Be sure to **always save the text outlines!** You don't need any other output for the purposes of this analysis.

## On a remote server

Reach out to your organization's computing cluster support to gain access to your cluster. With `anaconda` loaded in your `$PATH`, follow the instructions on the Cellpose GitHub page to install Cellpose exactly as you do locally, but without the GUI. You may have to use the steps to install `cudatoolkit` for Linux if you are unable to connect to GPU. The details of installing and running Cellpose on a remote cluster can be particular to your organization's infrastructure, so reach out to your cluster's support if you need help.

**In the command line or a script** With Cellpose installed in an Anaconda environment, it's possible to run Cellpose directly from the command line or a job script. There is a breakdown of command line commands here. I've found that Cellpose sometimes struggles with `--diameter` set to 0 for automatic diameter estimation, so play around to find a good value. This may take some trial and error to find a diameter that consistently segments well. The other important part is to use the `--save_txt` flag to save the text outlines.

You can also write these commands into a job script for batch segmentation. The job submission guidelines will be particular to your organization's job manager. Create your `conda` environment as before. Then, use `vim` to create `cellpose.sh`:

```
$ vim cellpose.sh
```

You can then enter the following (a copy is available in the scripts directory):

```
#!/bin/bash

conda activate cellpose

# Edit the parameters & paths below for your segmentation
# Descriptions of parameters and additional flags are available in the Cellpose documentation
python -m cellpose \
    --verbose \
    --use_gpu \
    --dir </path/to/images/> \
    --pretrained_model cyto2 \
    --chan 0 --chan2 0 \
    --diameter 0 \
    --save_txt \
    --no_npy
```

Look at the `cellpose` documentation for more information on how to run Cellpose in the command line. For a grayscale cytoplasm or nuclear image with no other color, `--chan` and `--chan2` should both be 0.

# Image analyses

## Dot detection with RS-FISH

### Overview

This pipeline is designed to provide single-cell resolution gene expression data to the qualitative, visual data generated by v3 HCR-FISH. The FIJI plugin RS-FISH is used to identify dots, which are turned into single-pixel maxima.

### Image pre-processing

After imaging you should have already processed the direct images with AiryScan processing, stitching, and max intensity projection. Then you will need to divide your processed whole-tissue, multi-channel image to separate out tissue of interest (blastema, epithelium, and mesenchyme, for example) as well as split the channels to run RS-FISH on. The instructions for doing this are here.

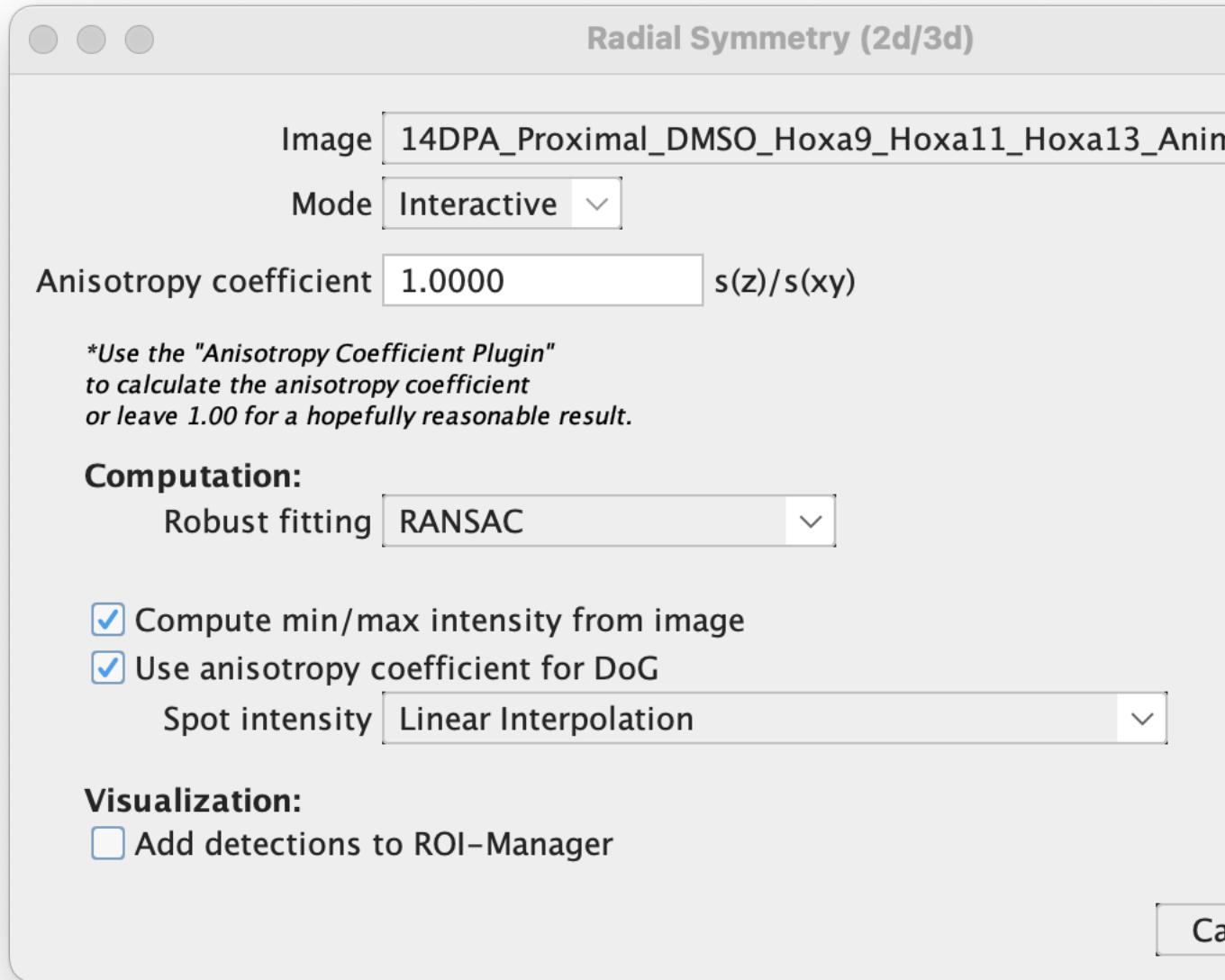
### RS-FISH FIJI plugin to identify dots

The RS-FISH FIJI plugin identifies dots in each of the fluorescent channels from your FISH images. By now, you should have a `.tif` for each of these channels, as well as a cytoplasm and/or nuclear channel. The instructions to install the RS-FISH plugin are available on the GitHub page.

Once RS-FISH is installed, open one of your fluorescent images. For the sake of this tutorial, it would be best to open one that has decently high expression and signal. If your image looks like the one below, this is not a problem - RS-FISH will still be able to detect dots. For ease of setting parameters and determining if the dots are reasonable, it's okay to adjust the brightness and contrast (**without** hitting **Apply**, which will change the pixel intensity values). Do this by going to **Image > Adjust > Brightness/Contrast...** and use the slider bars or **Set** to find good values.

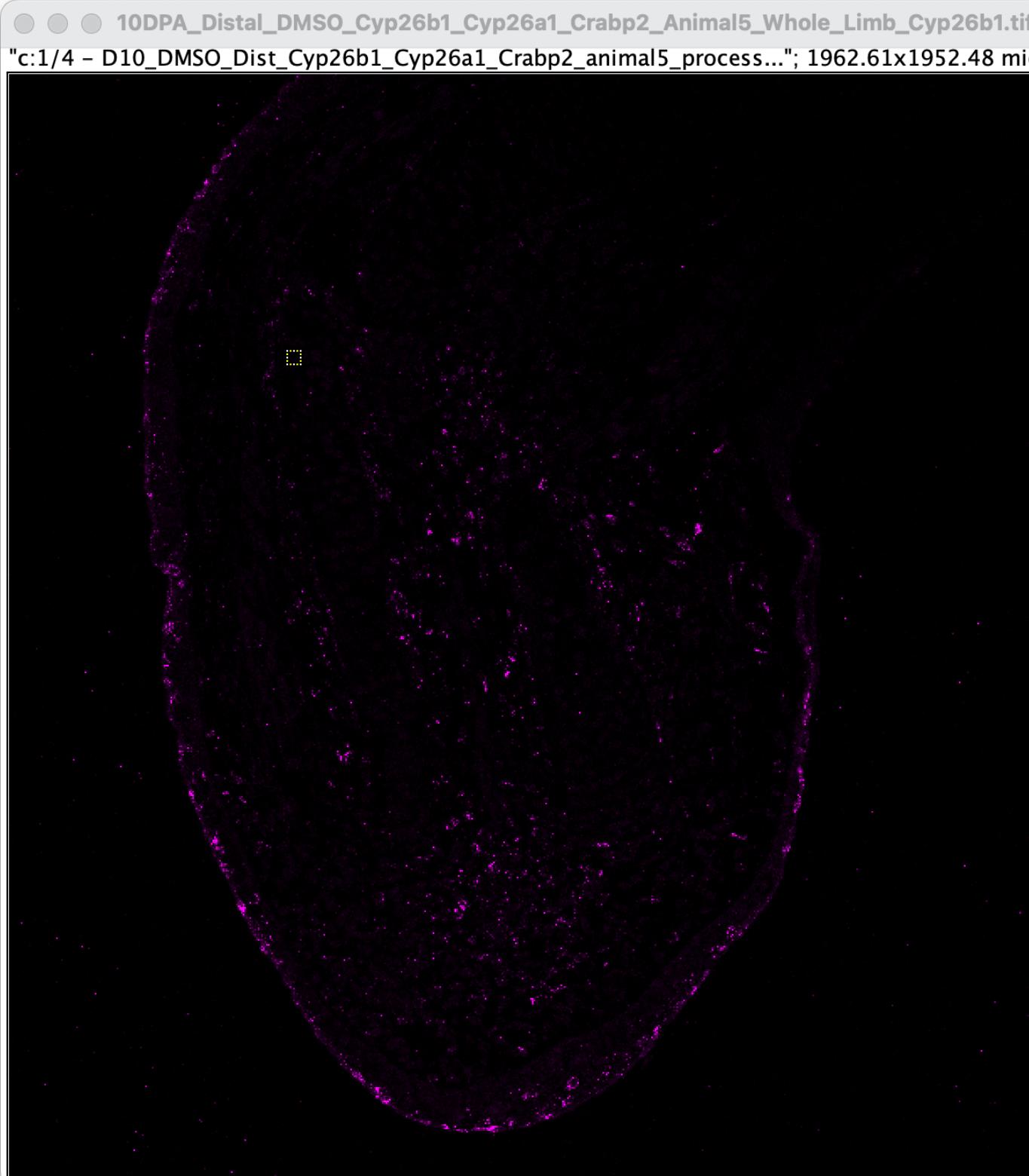


**Setting parameters in interactive mode** With your image open, go to Plugins > RS-FISH > RS-FISH. This will open a box titled Radial Symmetry (2d/3d):



For right now, everything on this screen can be left as-is, except that you should tick the box that says **Add detections to ROI-Manager**. If you have multiple images open, make sure the **Image** drop-down box is displaying the file you intend to work with. Hit **OK**.

A few new windows will pop up. One is your **Log**, which you can ignore for right now. Another is titled **Adjust difference-of-gaussian values**, and a third says **Adjust RANSAC values**. The final one is the same size as your input image, titled **RANSAC preview**. If you look at your original image, you'll see that, like in the **RANSAC preview**, there's a small rectangle drawn on:



This rectangle is a test area where the predicted dots will show up as you adjust your parameters. If it's too small, or not in an area where there are enough dots to be useful, you can go into the main FIJI bar and select the `Rectangle` tool. You can zoom in and move around on your image to find a good area, then redraw the rectangle where you want. It will automatically adjust in the `RANSAC preview` window too. You may already see some dots being picked up in this window.

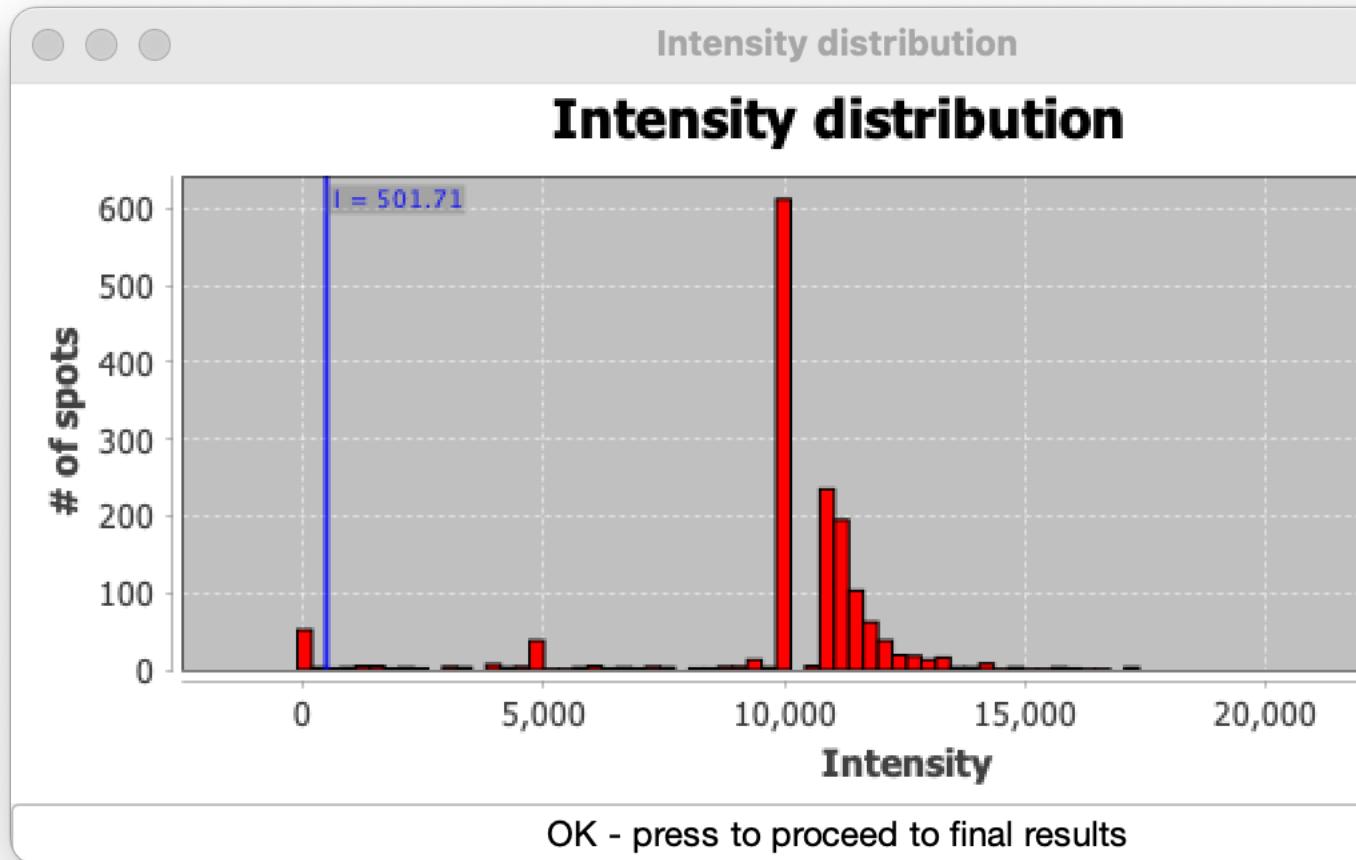
Now to dive into the parameters. You can read more about them and what they do on the RS-FISH Github page. They also walk through everything I've discussed above and the rest of what is covered regarding the plugin's use, so you can follow along their tutorial if you prefer.

In the `Adjust difference-of-gaussian values` window, you'll see `Sigma = 1.50` and `Threshold = 0.0070` with sliding bars underneath. These are arbitrary starting values, and will likely need to be adjusted.

- Sigma is essentially the expected size of a dot, and can change between channels or fluorophores. Use your judgment. Zoom in on your image to see how the diameter of detected dots changes as you slide the bar to find a value that seems fitting.
- Threshold is the cutoff for signal intensity using difference of Gaussian. As you slide the bar to the left (lower threshold), you'll notice more and more dots detected. Again use your judgment to find a value that looks like only real signal is being picked up, while background is ignored.

In the `Adjust RANSAC values` window, you'll see additional sliding bars and an option for background subtraction. I won't go into detail about these, but you can read about them in the RS-FISH documentation, and play around with them as you see fit. For now, it's also okay to leave them as-is, with `No background subtraction` selected. Hit `Done` in either of these boxes.

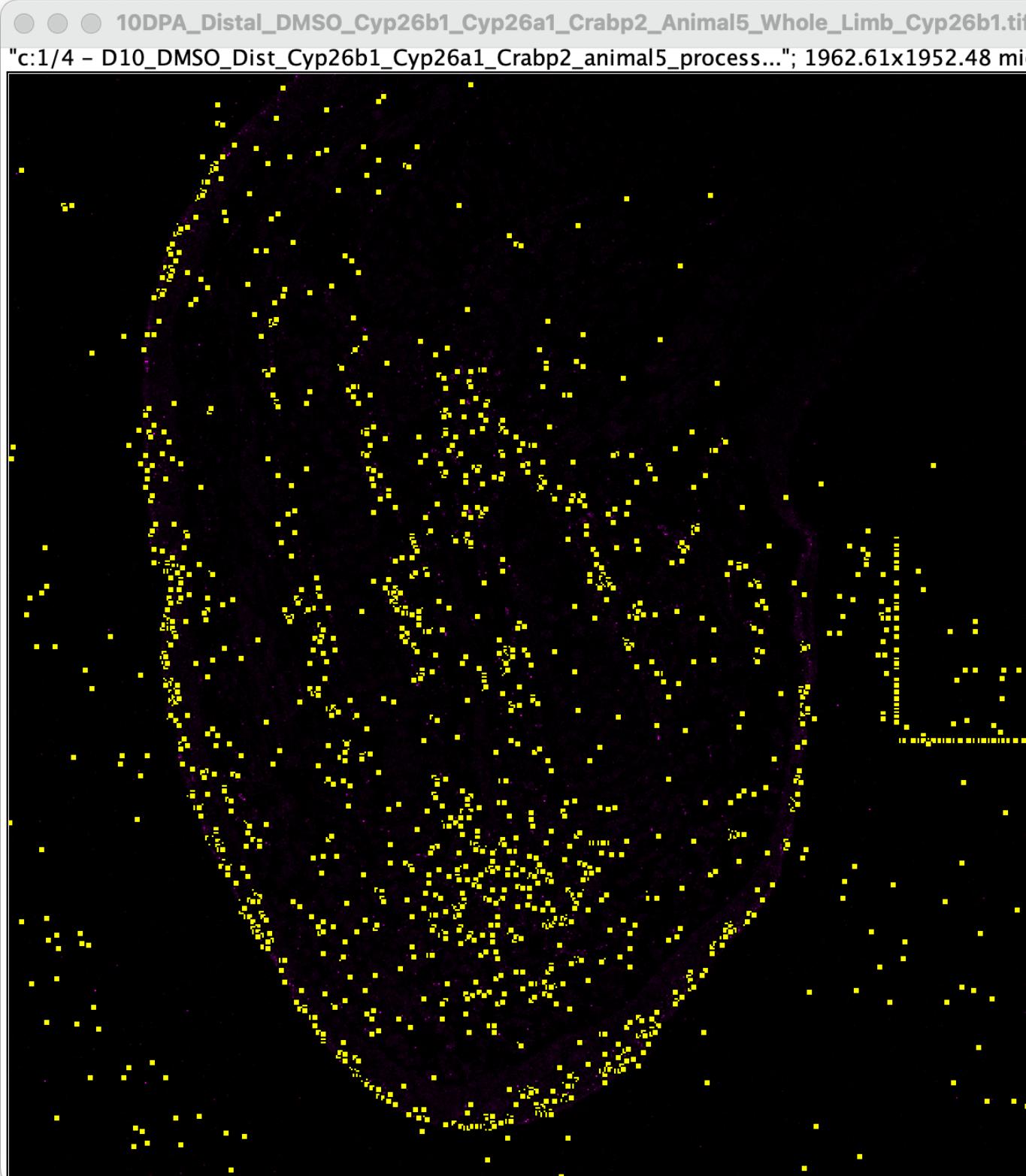
You'll see your `Log` start to run through the steps of RS-FISH, and a histogram will pop up like this:



The blue line is another threshold. If a dot's pixel intensity falls below that threshold, it gets excluded from the final results. Depending on your imaging setup, this threshold could vary quite a bit. We've found that best practice is to set this at 0 to avoid introducing subjectivity; however, potentially due to AiryScan processing, it seems that 10000 can actually be a “true” 0 value for many images. Set your cutoff at an appropriate value, then click **OK**.

RS-FISH will do its calculations reported in the **Log**, then two screens will pop up. One is a file of the detected dots' coordinates, which you can save if you want, but they won't be used further. The other is the **ROI Manager**. Here every dot is assigned an ROI number. If the ROI Manager doesn't pop up, you need to go back and tick the box that says **Add detections to ROI-Manager** in the RS-FISH opening screen before running RS-FISH again.

In the ROI Manager, click **Properties...**, then change **Point type:** to **Dot**. You can also change the default color if needed to aid visualization. Click **OK** and apply to all selections. Then check the box that says **Show All**, and the detected dots will show up overlaid on your image:



Zoom in and out, toggling **Show All** to assess whether your parameters were set well. If a lot of dots are being missed, or a lot of background is being picked up on, you'll need to go back and adjust them. If it looks good, you can move forward below.

**Dot detection in advanced mode** You can exit out of the ROI Manager and smFISH Localizations boxes from Interactive mode, discarding the dots that were found. Reopen RS-FISH with Plugins > RS-FISH > RS-FISH. In the screen that pops up, change the Mode from Interactive to Advanced. Also make sure Add detections to ROI-Manager is ticked. Hit OK.

A screen will open like this:

**Advanced Options**

Sigma (DoG)	0.78500
Threshold (DoG)	0.00185
Support region radius (RANSAC)	3
Min inlier ratio (RANSAC)	0.10
Max error (RANSAC)	1.50
Spot intensity threshold	9868.27

Background subtraction

Background subtraction max error	0.05
Background subtraction min inlier ratio	0.10

Results file

**Multi-Threading:**

*(Warning: if using RANSAC, results might slightly change from run to run due to inherent randomness)*

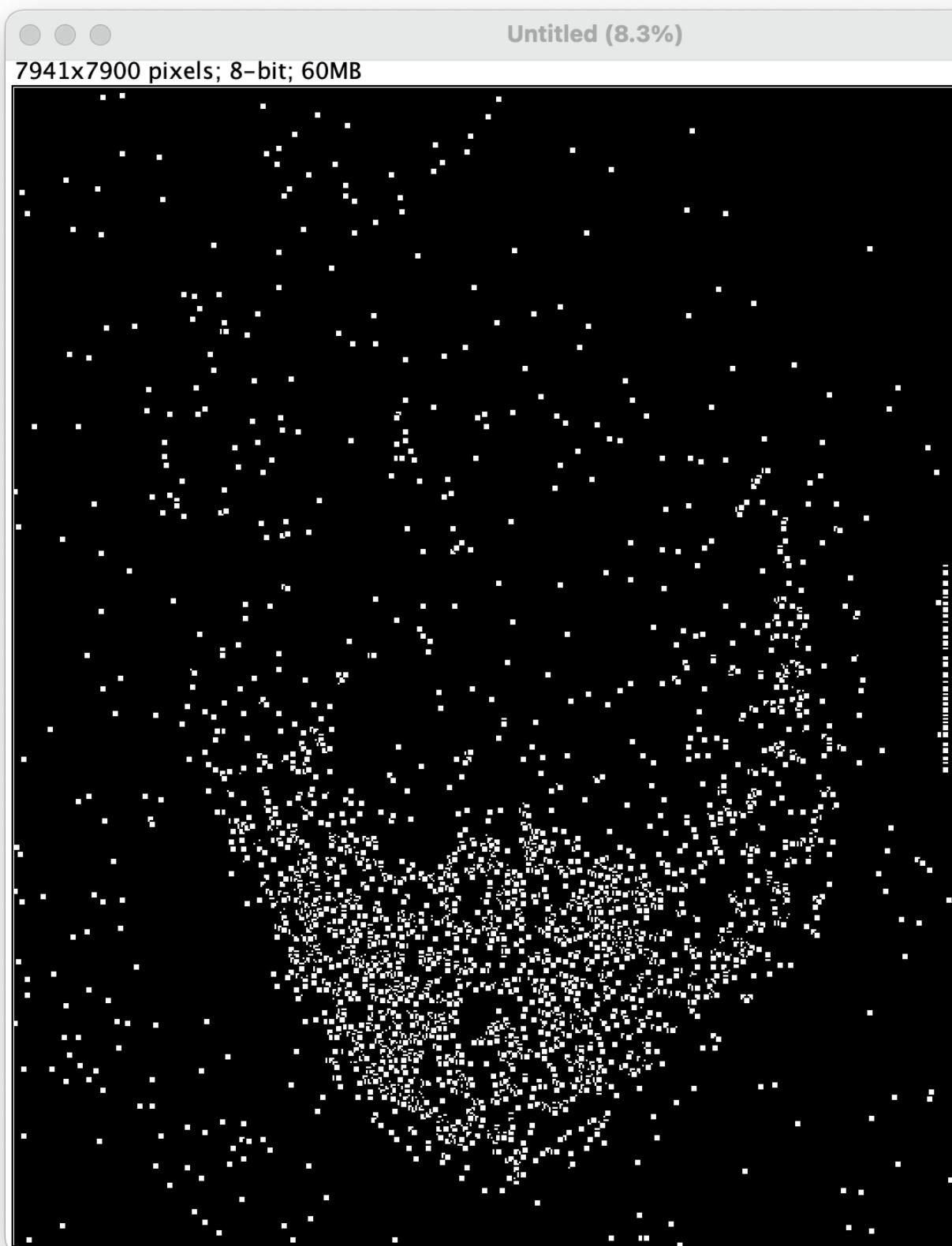
Use multithreading

Num threads	12
Block size X	128
Block size Y	128
Block size Z	16

If you came directly from **Interactive** mode without closing FIJI, this will be pre-populated with the most recent parameters you set. Here you can adjust as necessary, maybe making the **Spot intensity threshold** exactly 0 or 10000, and refining the other parameters as desired. Make a note of what values you set if you intend to use them on other images. Hit **OK**.

When the calculations are finished, the **ROI Manager** will open with your detected dots. Change the point type to **Dot**, and size to **Small**. The color doesn't matter, but my default is **white**. Now you can move on to the next step of making the Maxima image.

**Creating maxima image** Without closing the **ROI Manager**, go to **File > New > Image**. Make the **width** and **height** of the new image the same as the image you were just working with, and make **Fill with:** **black**. With this new blank image selected, tick **Show All** in the **ROI Manager**. The dots should overlay on the image like this:



In the ROI Manager window, hit **Flatten**. A new image will open, and it might look like nothing is there, but if you zoom in you'll see each of the dots are just smaller as they are no longer an overlay.

With the flattened image selected, go to **Process > Find Maxima**. Set your **Prominence** to 0. Change the **Output type** to **Single Points**, then hit **OK** and wait a second.

What will open is an image that will probably look blank, either fully white or fully black - it doesn't matter which. This is because the dots detected by RS-FISH have been reduced to single binary pixels, wherein a "positive" pixel (i.e., a dot) has a value of 255, and every other pixel (i.e., not a dot) has a value of 0. If you zoom in a lot, you will see that there are black (or white) pixels where the dots were detected. Save this image as a .tif with **\_Maxima.tif** at the end. In this example, **10DPA\_Distal-DMSO\_Cyp26b1-Cyp26a1-Crabp2\_Animal5\_Whole-Tissue\_Crabp2.tif** would become **10DPA\_Distal-DMSO\_Cyp26b1-Cyp26a1-Crabp2\_Whole-Tissue\_Crabp2\_Maxima.tif**.

This maxima image is what you will then use to count dots with an ROI converter.

### **Using the ROI converter to measure single-cell level expression**

This is the final step in collecting counts of dots from HCR-FISH images. Briefly, you will use the maxima images created in a previous step, then overlay your segmentation result outlines onto this image to get measurements of the dots in each nucleus or cell.

#### **Cellpose single-image ROI converter**

With your maxima image open in FIJI, make sure your ROI-Manager is cleared of any previous ROIs. You can do this simply by closing the ROI-Manager window if it is open. Open **File > New > Script...** and a window will open like this:

The image shows a screenshot of a code editor window. The title bar at the top right says "New\_.py". The main area contains a single line of code: "1". This line is highlighted with a yellow background. The code editor has a standard interface with scroll bars on the left and right sides. On the far right, there is a vertical toolbar or panel with several items: "Run", "Active", "Active", and "Autocom".

Go to **Language** and make sure **Python** is selected. Now go to the Cellpose Github page and copy-and-paste the ImageJ ROI Converter macro into this screen or download the file and open it in FIJI. Then click **Run**. If all goes smoothly, you'll be prompted to choose the text outlines pertaining to that image. These ROIs will automatically populate the ROI Manager and overlay onto your maxima image:



You can uncheck the **Labels** box in the ROI Manager to assess the segmentation results more easily. If the results don't look spectacular, you might want to go back and edit the segmentation parameters you set to get these outlines.

Go to **Analyze > Set Measurements....** You can measure whatever you want, but what we're interested in is **Label**, **Area** and **Integrated density**, so make sure at least those boxes are ticked, and hit **OK**. Then in the ROI Manager, hit **Measure**. A window will pop up with the measurements from that image, which you should save as a **.csv** corresponding to the original image, the gene, and the tissue.

For example, we might have **10DPA\_Distal-DMSO\_Cyp26b1-Cyp26a1-Crabp2\_Animal5\_Crabp2\_Blastema.csv**. The raw integrated density (**RawIntDen**) for each ROI indicates how many dots were in a given cell multiplied by 255. This is from the earlier **Find Maxima**, when each identified FISH dot became a single pixel with a value of 255 and "everything else" has a value of 0. You can ignore the **IntDen** column.

**Note: an infrequent error in ROI conversion** Occasionally when using the ROI converter, there will be an error something like this: `TypeError: invalid literal for int() with base 10: ''`. This is due to some strangeness in Cellpose where one or more empty lines are inadvertently added to the text file of outlines. To fix this error, you'll have to go through and manually delete the blank lines. This won't impact your segmentation. If you are on Mac or Linux, you can run the command below in your command line to print out the line numbers that are empty (first make sure you are in the folder containing your text outlines):

```
for FILE in *
do
    echo $FILE;
    awk '{if($0 == "") {print NR}}' $FILE
done;

## Cellpose_Segmentation.Rmd
## 11
## 15
## 18
## 21
## 24
## 28
## 31
## 34
## 36
## 39
## 42
## 45
## 50
## 54
## 56
## 69
## 71
## Cellpose_Segmentation.md
## 3
## 17
## 19
## 25
## 27
## 34
## 36
```

```
## 44
## 46
## 48
## 52
## 54
## 65
## 67
## 76
## 81
## 83
## 90
## 92
## 102
## 104
## 114
## 119
## 123
## 126
## 129
## 131
## 144
## 150
## 152
## 154
## 156
## 163
## 165
## 167
## 171
## 173
## 175
## 179
## 181
## 183
## 188
## 190
## Data_Concatenation_Walkthrough.Rmd
## 8
## 11
## 13
## 15
## 33
## 43
## 49
## 60
## 66
## 72
## 76
## 80
## 86
## 127
## 131
## 134
## 136
```

```
## 191
## 194
## 200
## 203
## 210
## 241
## FISH_Intensity_Measurement.Rmd
## 5
## 12
## 14
## 16
## 18
## 21
## 25
## 27
## 31
## 33
## 35
## 39
## 41
## 45
## 47
## 49
## 51
## 55
## 57
## FISH_Intensity_Measurement.md
## 3
## 7
## 14
## 19
## 21
## 26
## 28
## 35
## 37
## 41
## 44
## 46
## 49
## 51
## 53
## 61
## 65
## 67
## 71
## ImageAnalysis.Rmd
## 10
## 20
## 23
## 25
## 28
## 34
## 36
```

```
## 39
## 42
## 46
## 48
## 52
## 54
## 58
## 60
## 64
## 67
## 71
## 73
## 77
## 79
## 81
## 85
## 87
## 89
## 94
## 96
## 99
## 106
## 111
## 113
## 115
## 118
## 122
## 124
## 128
## 130
## 132
## 136
## 138
## 141
## 154
## 158
## 161
## 164
## 168
## 171
## 174
## 176
## 179
## 182
## 185
## 190
## 194
## 196
## 209
## 211
## 214
## 217
## 220
## 223
```

```
## 228
## 234
## 236
## 241
## 243
## 245
## 247
## 249
## 251
## 253
## 258
## 260
## 262
## 267
## 269
## 272
## 277
## 279
## 281
## 287
## 289
## 291
## 293
## 295
## 298
## 304
## 309
## 311
## 313
## 315
## 325
## 327
## 330
## 332
## 334
## 338
## 340
## 347
## 350
## 352
## 355
## 357
## 361
## 363
## 365
## 368
## 372
## 374
## 378
## 380
## 382
## 386
## 388
## 392
```

```
## 394
## 396
## 398
## 402
## 404
## 406
## ImageAnalysis.log
## 1
## ImageAnalysis.tex
## 108
## 112
## 115
## 121
## 124
## 126
## 128
## 130
## 132
## 138
## 143
## 145
## 153
## 155
## 164
## 166
## 179
## 181
## 191
## 193
## 201
## 203
## 206
## 214
## 216
## 220
## 222
## 238
## 242
## 244
## 251
## 254
## 266
## 269
## 272
## 284
## 286
## 288
## 297
## 299
## 312
## 329
## 331
## 340
## 342
```

```
## 352
## 354
## 357
## 366
## 368
## 371
## 373
## 375
## 384
## 390
## 399
## 411
## 420
## 431
## 433
## 435
## 441
## 443
## 450
## 453
## 461
## 463
## 465
## 469
## 472
## 483
## 485
## 494
## 499
## 501
## 508
## 510
## 520
## 523
## 533
## 538
## 544
## 547
## 551
## 553
## 567
## 574
## 576
## 579
## 581
## 586
## 588
## 595
## 598
## 605
## 616
## 618
## 621
## 625
```

```
## 627
## 633
## 641
## 643
## 652
## 658
## 663
## 677
## 684
## 687
## 689
## 698
## 706
## 712
## 714
## 719
## 722
## 730
## 732
## 734
## 742
## 748
## 750
## 757
## 759
## 763
## 768
## 780
## 783
## 786
## 792
## 795
## 801
## 803
## 812
## 814
## 819
## 827
## 835
## 838
## 849
## 859
## 1371
## 1374
## 1377
## 1386
## 1394
## 1397
## 1413
## 1415
## 1420
## 1428
## 1435
## 1440
```

```
## 1448
## 1458
## 1460
## 1465
## 1467
## 1470
## 1472
## 1476
## 1483
## 1489
## 1491
## 1501
## 1503
## 1517
## 1519
## 1532
## 1534
## 1543
## 1545
## 1566
## 1568
## 1584
## 1586
## 1588
## 1590
## LICENSE
## 2
## 4
## 11
## 14
## README.md
## 2
## 5
## 13
## 15
## 18
## 21
## 23
## 25
## 27
## 29
## ROI_Converters.Rmd
## 11
## 15
## 18
## 24
## 29
## 31
## 33
## 35
## 45
## 47
## 50
## 52
```

```
## 54
## 58
## 60
## 67
## 70
## 72
## 75
## ROI_Converters.md
## 3
## 12
## 14
## 20
## 22
## 28
## 37
## 42
## 49
## 57
## 59
## 70
## 78
## 81
## 83
## 92
## 100
## 103
## 112
## 114
## 119
## 124
## 127
## 137
## 139
## 145
## 147
## 149
## 151
## 158
## 160
## 162
## 166
## 168
## 170
## 176
## 178
## 180
## 185
## 187
## RS_FISH_Analysis.Rmd
## 11
## 15
## 18
## 21
## 24
```

```
## 29
## 35
## 37
## 42
## 44
## 46
## 48
## 50
## 52
## 54
## 59
## 61
## 63
## 68
## 70
## 73
## 78
## 80
## 82
## 88
## 90
## 92
## 94
## 96
## RS_FISH_Analysis.md
## 3
## 13
## 15
## 20
## 22
## 29
## 31
## 37
## 48
## 50
## 54
## 59
## 67
## 75
## 82
## 87
## 92
## 98
## 105
## 109
## 117
## 125
## 131
## 136
## 138
## 144
## 147
## 154
## 159
```

```
## 161
## 168
## 172
## 176
## 188
## 191
## 193
## 195
## 197
## 203
## 205
## 207
## 214
## 216
## Tissue_Isolation.Rmd
## 11
## 15
## 18
## 20
## 23
## 26
## 30
## 32
## 36
## 38
## 42
## 44
## 48
## 51
## 55
## 57
## 61
## 63
## 65
## 69
## 71
## 73
## 78
## 80
## 83
## 90
## 95
## 97
## 99
## 102
## 106
## 108
## 112
## 114
## 116
## 120
## 122
## 125
## 137
```

```
## Tissue_Isolation.md
## 3
## 14
## 16
## 22
## 27
## 29
## 37
## 39
## 43
## 45
## 51
## 53
## 58
## 60
## 68
## 70
## 72
## 74
## 76
## 80
## 82
## 91
## 95
## 97
## 103
## 106
## 111
## 114
## 116
## 122
## 125
## 129
## 135
## 146
## 148
## 151
## 153
## 156
## 158
## 161
## 164
## 166
## 169
## 171
## 173
## 177
## 179
## 182
## 187
## 190
## 194
## 196
## 198
```

```

## 200
## 207
## 209
## 211
## 217
## 219
## Untitled.rtf
## 8
## img
## references.bib
## 15
## 30
## 45
## 60
## 75
## scripts

```

Then you can easily go into a text editor like Vim and find and delete the empty lines.

### Running the batch ROI converter

The individual conversion above is useful for checking on your segmentation parameters, as well as assessing if any “iffy” images segmented adequately enough for their measurements to be valid. However, as your numbers grow it can get a little tiresome, so the Cellpose-provided ROI converter was modified a little bit to be able to work through whole folders of maxima and outlines and automatically take the measurements. A copy is provided in the scripts directory.

When you run the batch converter, a window will pop open prompting you to select input and output directories. The input directory should be the directory containing both the `_Maxima.tif` images and `_cp_outlines.txt` segmentation text outlines. The output directory should be an empty (or not, but it will overwrite any existing results files if they match in name) directory where all of the measurements files will populate.

There are a few things in the batch converter that you will need to edit for your personal use:

- At line 13, change the `tissues` list to reflect your (- separated single-string) regions of interest. This might instead be something like ("Ventral-Mesenchyme", "Distal", "Proximal-Nail").
- At line 15, the `stain` should be the one used for the given image and outlines, such as WGA or GFP. If your stain is multiple words, again connect them with a -.
- If you are running on Windows, you may need to change the / in line 47 to \.

Some things to make sure of before running the batch converter:

- All maxima images should be named with this format:  
– `{Other_Identifying_Information}_Whole-Tissue_{Gene}_Maxima.tif`
- All outline text files should be named with this format:  
– `{Other_Identifying_Information}_{Tissue}_{Stain}_cp_outlines.txt`

In our case, we have two separate conditions, days post amputation and a drug treatment. We also do three-gene FISH with a DAPI nuclear stain, and have multiple biological replicates. So an example for ours might be:

- 14DPA\_Distal-DMSO\_Hoxa9-Hoxa11-Hoxa13\_Animal1\_Whole-Tissue\_Hoxa9\_Maxima.tif
- 14DPA\_Distal-DMSO\_Hoxa9-Hoxa11-Hoxa13\_Animal1\_Blastema\_DAPI\_cp\_outlines.txt

This pairing will measure the dots/transcripts of Hoxa9 that were identified specifically in the limb blastema. The batch converter loops through your directory of maxima images and text outlines, matches each outlines file to its respective maxima image, and takes the measurements. It then outputs a .csv results file in your selected output folder that is named to reflect the input maxima and outlines. With the above maxima and outlines, for example, the batch converter will output 14DPA\_Distal-DMSO\_Hoxa9-Hoxa11-Hoxa13\_Animal1\_Hoxa9\_Blastema.csv.

## What to do now

After getting all of your measurement files, you are ready to put all of your data together for visualization and charting. This is done entirely in R, and a walkthrough is provided. You will need to download this file and open it in RStudio.

## Statistics

### Expression intensity along the proximodistal axis

#### Overview

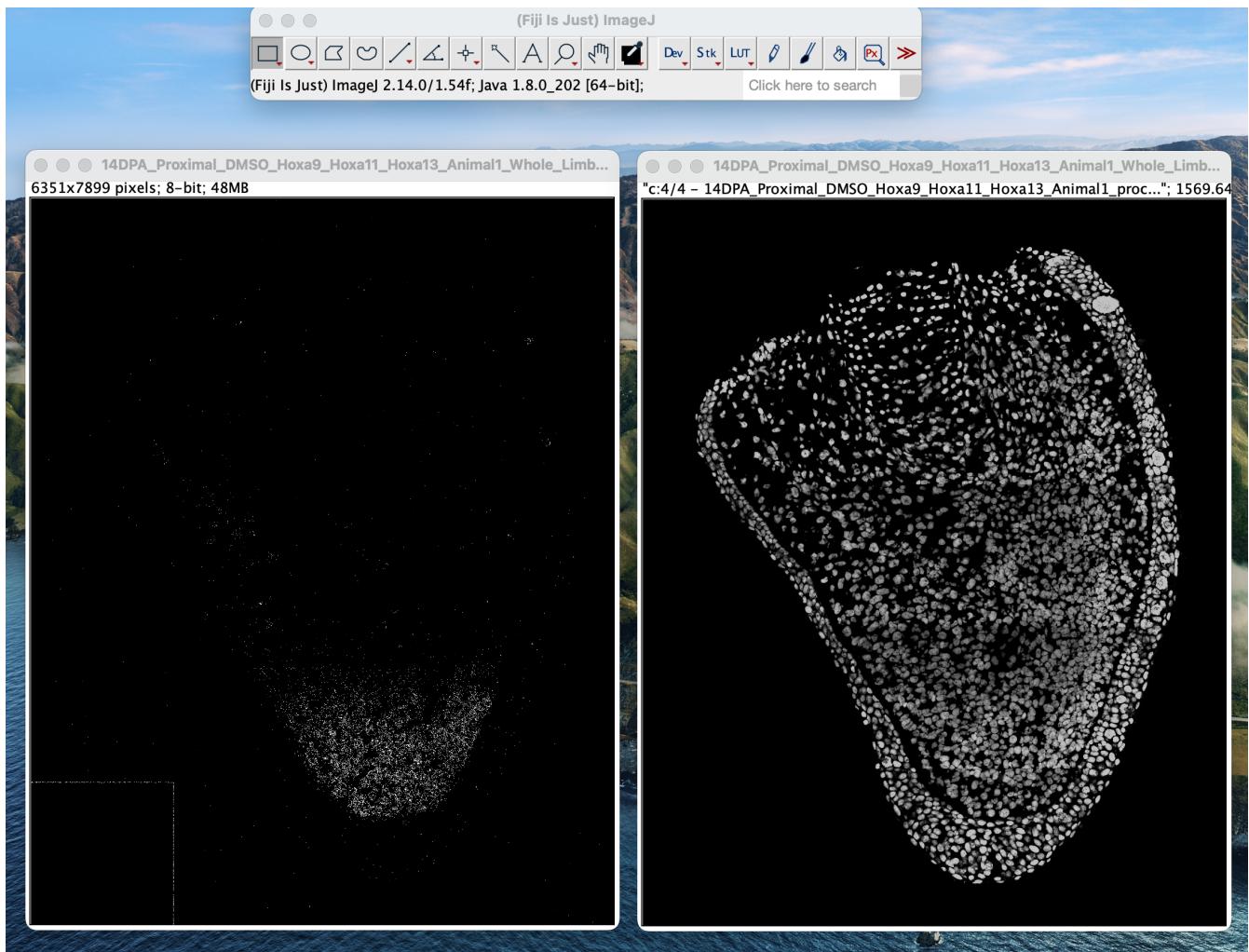
The goal of this pipeline is to get expression intensity measurements along the proximo-distal axis in the mesenchyme using HCR-FISH dots identified as single “positive” pixels.

**Some notes:** Depending on exactly the image size, each image may take up to 10 minutes and produce an output CSV over 1 GB. Do not start saving the results CSV until the macro is entirely finished running, as the results page will appear before it's actually finished. Do not close out of the results page until it is finished saving - because these are such large results files, it will take a few minutes.

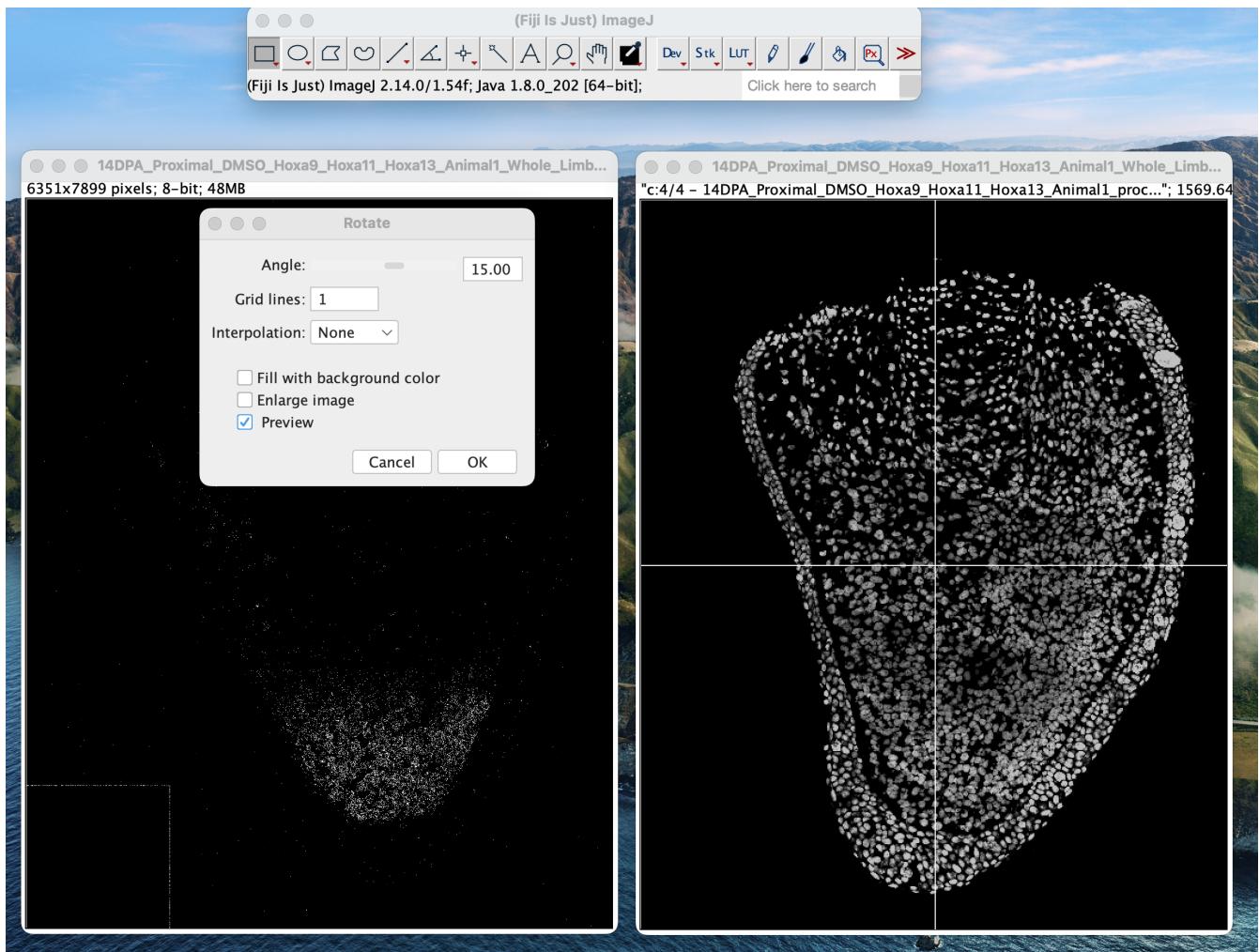
Be sure to name your output files along the lines of {Timepoint}\_{Condition}\_{Gene}\_{Replicate}.csv, which aligns with the naming convention default in the provided plotting script.

#### Steps

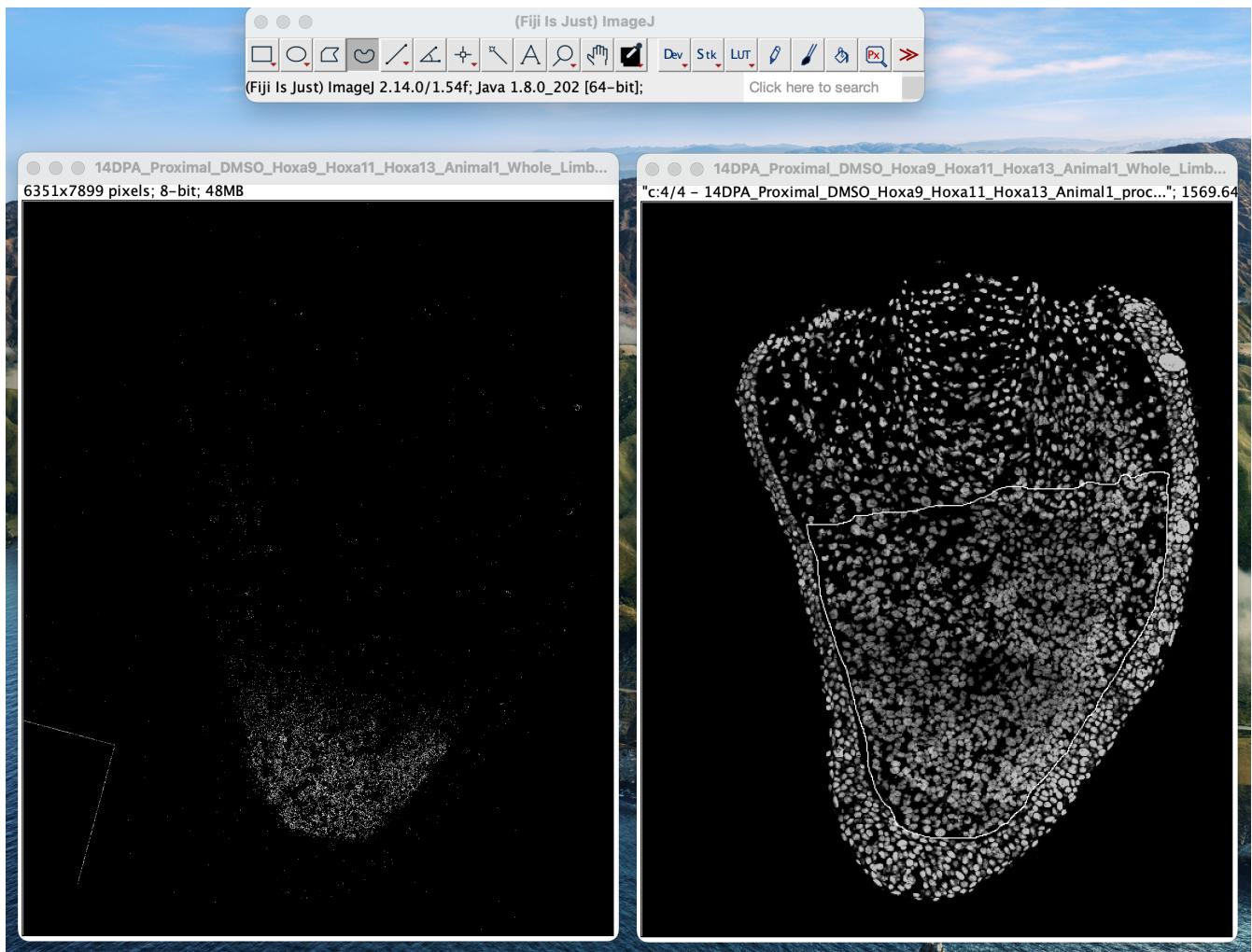
1. In FIJI, open both the maxima image (generated here) and corresponding DAPI image. Note that the maxima here have been dilated for visibility.



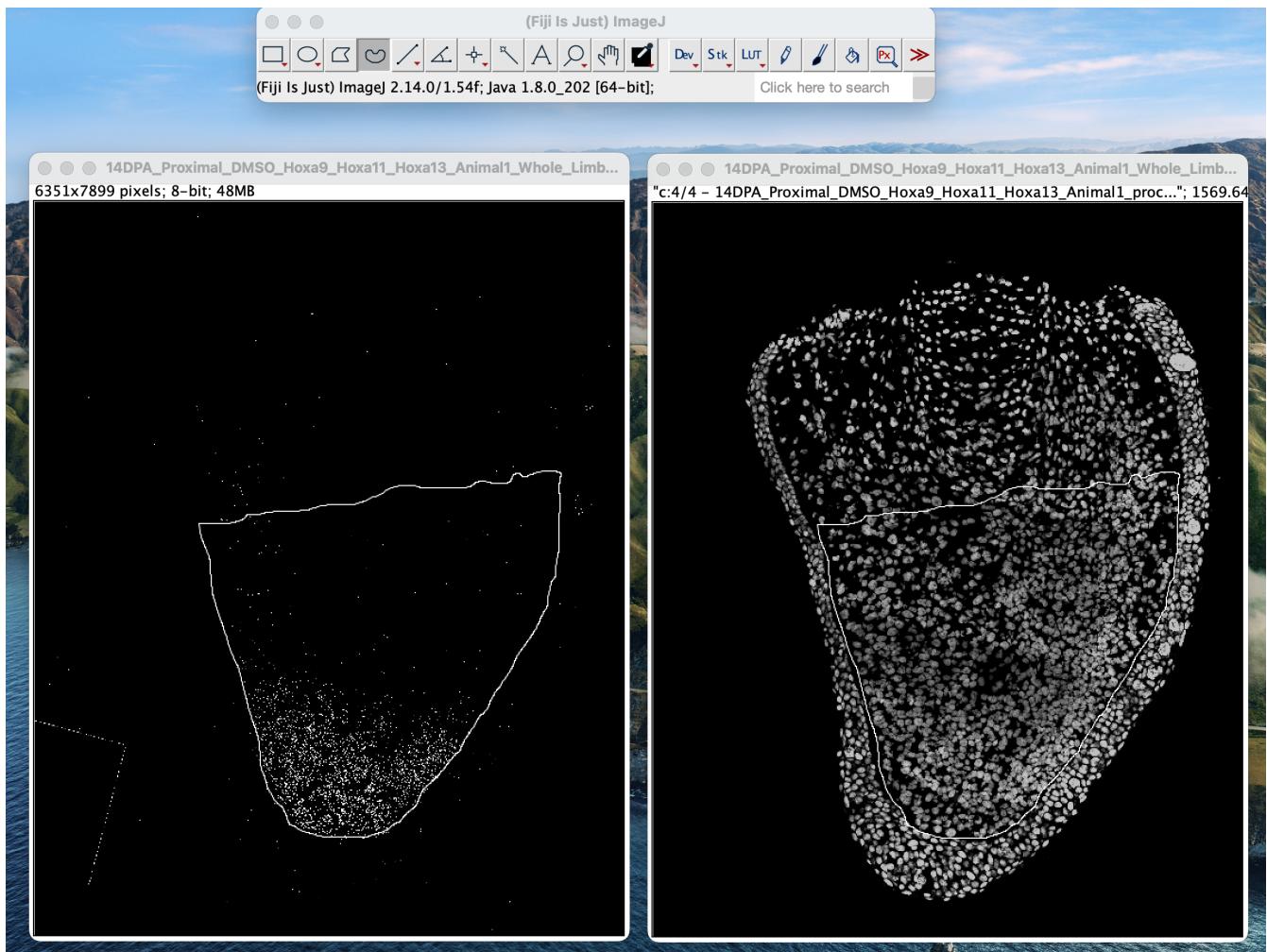
2. Visually assess the directionality of the tissue using the DAPI image. The proximo-distal normalization is based on the y-axis, so if the tissue is very “tilted” then it may need to be adjusted. Go to **Image > Transform > Rotate...** and a screen will pop up. Tick the **Preview** box. Change the value in the **Angle** box until it is roughly proximal at the top, distal at the bottom.



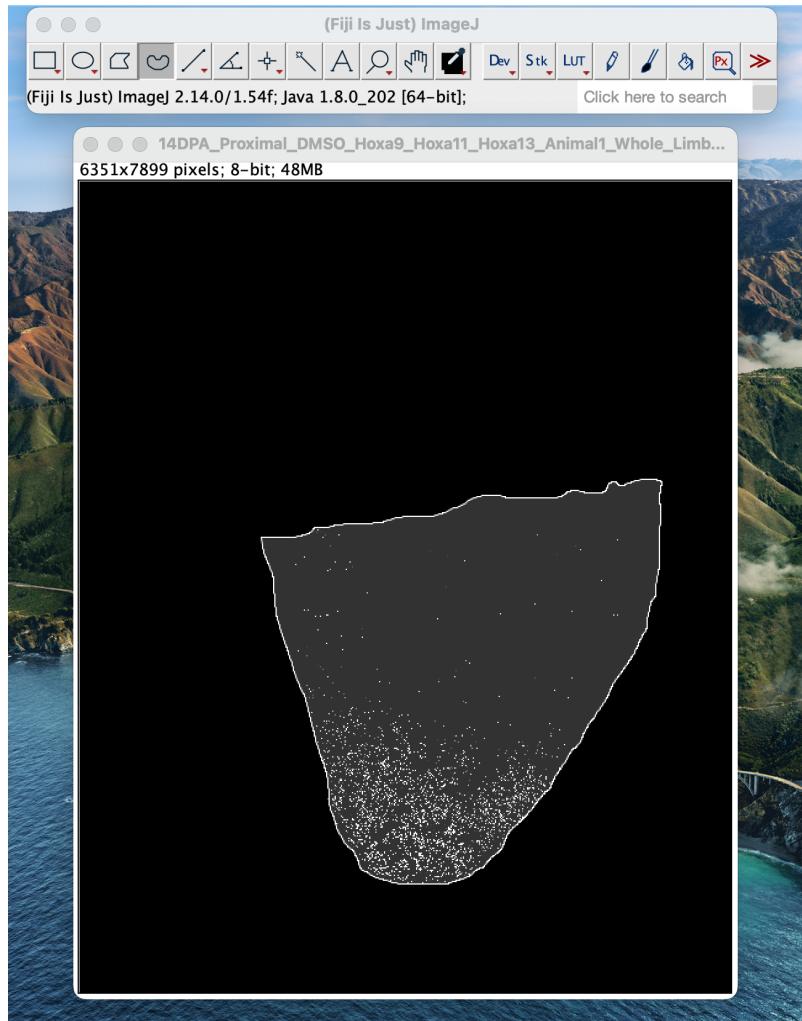
3. Make note of the angle chosen, and press **OK**. Then repeat the same rotation on the maxima image.  
**NOTE:** in the **Rotate** box that opens, set **Interpolation** to **None**.
4. Again in the DAPI image, use the freehand tool to outline the tissue of interest:



5. Select the maxima image, then go to Edit > Selection > Restore Selection.



6. In the maxima image, go to **Edit > Clear Outside**.
7. Use **Process > Math > Add** and add an arbitrary value greater than 0 and less than 255 to the selection area. It doesn't really matter what this value is, as long as it's consistent between images. This is how the plotting script distinguishes between "in the selection" negative pixels and "outside of the selection" pixels. Anything that was 0 will now be the added value, while the positive pixels will remain at the maximum value, 255.
8. After step 7 the image should look something like this, which may be darker or lighter inside the selection depending on the value added in the previous step:



9. Run the intensity measurement macro on this image. It will take a few minutes to finish.
10. When the macro finishes running, select the results page and save it as a .csv. Give it a few minutes to finish saving. The naming convention {Timepoint}\_{Condition}\_{Gene}\_{Replicate}.csv is used in the plotting script provided.

## Colocalization

### Statistics

### References